REMARKS/ARGUMENTS

In view of the foregoing amendments and the following remarks, reconsideration of this application is requested. Claims 7-28 are now pending with claims 7, 11, 18, and 22 being independent. Claims 1-6 have been canceled. New claims 18-28 have been added. No new matter has been introduced.

The title of the application at page 2 and cross reference to related applications at pages 2 and 3 have been amended as requested by the Examiner. Applicants have amended the specification on page 11, line 4 as requested by the Examiner. Finally, Applicants have checked the specification for minor errors.

Claim 7 describes a method, comprising fetching and decoding instructions in a first processor, detecting an unsupported instruction that is not executable by the first processor, executing said unsupported instruction in a second processor, and providing the first processor with a supported instruction that is executable in the first processor without the first processor fetching said instruction.

Claim 11 describes a system, comprising a first processor having fetch logic and decode logic, the first processor fetches supported instructions from memory using said fetch logic and decodes said supported instructions with said decode logic, a second processor, the second processor executes unsupported instructions, means for loading a supported instruction in said decode logic of the first processor so that the first processor decodes but does not fetch the supported instruction, means for coordinating when said loading a supported instruction in said decode logic of the first processor so that the first processor decodes but does not fetch the supported instruction occurs.

Independent claims 7 and 11 stand rejected under 35 U.S.C. § 102(e) as being anticipated by Park et al. (U.S. Patent No. 6,832,305). Applicants request reconsideration and withdrawal of this rejection for at least the reason that Park does not describe or suggest in claim 7 providing the first processor with a supported instruction that is executable in the first processor without the first processor fetching said instruction. Park does not describe or suggest in claim 11 means for loading a supported instruction in said decode logic of the first processor so that the first processor decodes but does not fetch the supported instruction.

Park in col. 4, lines 31-48, referring to Figure 1, describes the 16 'm' bit instruction fetched from MPM 140 is sent to CPU 110, first into fetch buffer 111. At the output of fetch buffer 111, the 3 'c' MSB bits are branched off the predecoder 113, where if the 'c' bits signal a coprocessor operation, signal COPI 116 and CCLK 119 are activated. Thirteen of the 16 m bits (m-c) are branched to coprocessor 120. CCLK 119 clock into the coprocessor instruction register 121 the entire 29 bits of coprocessor instruction, with 13 bits from the output of CPU instruction fetch buffer 111 and 16 'n' bits from CPM 160 through coprocessor instruction fetch buffer 126. Thus, the coprocessor fetches 'n' bits of the supported instruction to form an 13+n bit coprocessor instruction that is decoded and executed by the coprocessor. Figure 1 shows Coprocessor Instruction Fetch Buffer 126 for fetching part of the coprocessor instruction. Park does not describe or suggest in claim 7 providing the first processor with a supported instruction that is executable in the first processor without the first processor fetching said instruction. Park does not describe or suggest in claim 11 means for loading a supported instruction in said decode logic of the first processor so that the first processor decodes but does not fetch the supported instruction.

For at least the reasons given above, Applicants respectfully submit that claims 7 and 11 are patentable over Park.

Claims 8-10 depend from independent claim 7 and claims 12-17 depend from independent claim 11. Accordingly, Applicants request reconsideration and withdrawal of the rejections for claims 8-10 and 12-17 for the reasons discussed above with respect to claims 7 and 11.

Dependent claim 17 stands rejected under 35 U.S.C. §103(a) as being unpatentable over Park et al. (U.S. Patent No. 6,832,305). Park does not describe or suggest all of the limitations of claim 11 as described above. Because claim 17 depends from claim 11, Applicants request reconsideration and withdrawal of the rejections for claim 17 for the reasons discussed above with respect to claim 11.

Claim 13 stand rejected under 35 U.S.C. § 103(a) as obvious over Park in view of Chaudhry et al., U.S. Pat. No. 6,681,318 ("Chaudhry"). Applicants respectfully submit that neither Park nor Chaudhry, whether considered singly or in combination, teach or suggest all of the limitations in claim 13. Applicants offer the following comments in support of their position.

Establishing a case of prima facie obviousness requires that all of the claim limitations must be suggested or taught in the prior art references. *In re Royka*, 180 U.S.P.Q. 580 (C.C.P.A. 1974). For at least the reasons set forth above in Applicants remarks concerning the rejection of claim 11 under 35 U.S.C. § 102(e) as being anticipated by Park, Applicants submit that neither Park nor Chaudhry whether considered singly or in combination, teach or suggest all of the limitations of claim 11 from which claim 13 depends. In particular, none of the cited references whether considered singly or in combination, teach or suggest "means for loading a supported instruction in said decode logic of the first processor so that the first processor decodes but does not fetch the supported instruction" as recited in claim 11. Consequently, the cited references cannot teach all of the limitations of claim 13.

New claim 18 describes a method, comprising fetching and decoding instructions in a first processor, detecting an unsupported instruction that is not executable by the first processor, executing said unsupported instruction in a second processor, providing the first processor with a supported instruction that is executable in the first processor without the first processor fetching said instruction, and detecting patterns of supported and unsupported instructions yet to be executed.

Neither the Park nor Chaudhry reference describe or suggest providing the first processor with a supported instruction that is executable in the first processor without the first processor fetching said instruction and detecting patterns of supported and unsupported instructions yet to be executed. Park in column 4, lines 1-48 recites "Predecoder 113 of CPU 110 receives the `c` MSB bits of the main program instruction and predecodes the instruction type code of the fetched instruction prior to decoding of the whole instruction by decoder 170 of CPU 110. If the instruction type code indicates a coprocessor instruction, the Predecoder 113 generates a Coprocessor Instruction Signal (COPI) 116. ... is used as the clock signal for synchronously clocking the coprocessor instruction into coprocessor 120. ... coprocessor 120 receives a complete coprocessor instruction comprising MSBs fetched by the CPU 110 and LSBs fetched by coprocessor 120. " No part of the Chaudhry reference describes or suggests detecting patterns of supported and unsupported instructions yet to be executed. Thus, neither Park nor Chaudhry describe or suggest detecting patterns of supported and unsupported instructions yet to be executed.

New claim 22 describes a system, comprising a first processor having fetch logic and decode logic, the first processor fetches supported instructions from memory using said fetch logic and decodes said supported instructions with said decode logic, a second processor, the second processor executes unsupported instructions, means for loading a supported instruction in said decode logic of the first processor so that the first processor decodes but does not fetch the supported instruction, means for coordinating when said loading a supported instruction in said decode logic of the first processor so that the first processor decodes but does not fetch the supported instruction occurs comprises a control program that examines patterns of supported and unsupported instructions yet to be executed and causes the first processor to switch between multiple instruction execution modes according to the patterns.

Neither the Park nor Chaudhry reference describe or suggest means for coordinating when said loading a supported instruction in said decode logic of the first processor so that the first processor decodes but does not fetch the supported instruction occurs comprises a control program that examines patterns of supported and unsupported instructions yet to be executed and causes the first processor to switch between multiple instruction execution modes according to the patterns. Park in column 4, lines 1-48 recites "Predecoder 113 of CPU 110 receives the `c` MSB bits of the main program instruction and predecodes the instruction type code of the fetched instruction prior to decoding of the whole instruction by decoder 170 of CPU 110. If the instruction type code indicates a coprocessor instruction, the Predecoder 113 generates a Coprocessor Instruction Signal (COPI) 116. ... is used as the clock signal for synchronously clocking the coprocessor instruction into coprocessor 120. ... coprocessor 120 receives a complete coprocessor instruction comprising MSBs fetched by the CPU 110 and LSBs fetched by coprocessor 120. " No part of the Chaudhry reference describes or suggests a control program that examines patterns of supported and unsupported instructions yet to be executed and causes the first processor to switch between multiple instruction execution modes according to the patterns. Thus, neither Park nor Chaudhry describe or suggest a control program that examines patterns of supported and unsupported instructions yet to be executed and causes the first processor to switch between multiple instruction execution modes according to the patterns.

In view of these remarks and amendments, Applicants submit that this application is now in condition for allowance and the Examiner's prompt action in accordance therewith is

respectfully requested. The Commissioner is authorized to charge any additional fees and/or credit any overpayment to Deposit Account 20-0668 of Texas Instruments Incorporated.

Respectfully submitted,

Indranil Chowdhury
Attorney for Applicants
Reg. No. 47,490

Ron Neerings
Texas Instruments Incorporated
P.O. Box 655474, MS 3999
Dallas, TX 75265
(972) 917-5290